Jenkins Practical Assignment

Objective: Implement a fully functional Jenkins environment and demonstrate practical scenarios using Jenkins pipelines and jobs.

Assignment Tasks

Part 1: Jenkins Setup

1. Set up Jenkins on a Linux VM and ensure it is functional and secure.
2. Install the following plugins:
   - Eclipse Temurin Installer
   - Pipeline Stage View

Part 2: Freestyle Job

1. Create a Freestyle Job for a Maven project.
2. Configure the job to build the project successfully.

Part 3: Declarative Pipelines

1. **Generic Webhook Trigger**
   Create a pipeline that triggers automatically upon code push via a webhook.
2. **Use of Parameters**
   Implement a pipeline that uses parameters to control builds for different environments (e.g., Dev, QA, Prod).
3. **Upstream and Downstream Jobs**
   Set up an upstream job that triggers a downstream job upon successful execution.
4. **Shared Libraries**
   Use Jenkins shared libraries to include reusable functions in your pipeline.
5. **Multibranch Pipeline with Webhook**
   Configure a multibranch pipeline that triggers specific branch builds using webhooks.
6. **Execute Stage Based on File Change**
   Implement conditional execution of a pipeline stage if a specific file is modified in the repository.

7. **Post Actions for Cleanup**
   Add post-build actions to clean up the workspace after job execution.
8. **Parallel Stage Executions**
   Create a pipeline that runs multiple stages (e.g., Unit Tests, Integration Tests, UI Tests) in parallel.
9. **Handling Long-Running Stages with Timeouts**
   Add a timeout to a long-running stage to terminate it if it exceeds a specified duration.
10. **Sending mail Notifications to Outlook From Jenkins**

Expected Deliverables

- Fully functional Jenkins instance with the required plugins installed.
- A working Freestyle Job for Maven project builds.
- Declarative pipelines demonstrating:
  - Generic webhook trigger
  - Use of parameters
  - Upstream and downstream job integration
  - Shared libraries
  - Multibranch pipeline with webhook
  - Conditional execution based on file changes
  - Post-build cleanup actions
  - Parallel stage executions
  - Timeout handling for long-running stages
  - Sending mail Notifications to Outlook From Jenkins

Submission: Provide screenshots showcasing the Jenkins setup, jobs, and pipeline runs. Include pipeline code snippets where applicable.